

---

TFC - Desenvolupament d'aplicacions per dispositius mòbils

**.Castells**

Memòria final

Autor: **Josep A. Gabriel Massana**  
Enginyeria Tècnica en Informàtica de Gestió  
Consultor: **Jordi Ceballos Villach**

10 de juny de 2012

*A la persistència, la confiança i el futur.*

## Resum

---

Aquesta memòria representa un resum final del projecte .Castells desenvolupat com a Treball Final de Carrera dins de l'àrea Desenvolupament d'aplicacions per dispositius mòbils dels estudis d'Enginyeria Tècnica Informàtica de Gestió de la Universitat Oberta de Catalunya.

El projecte .Castells permet a qualsevol persona que participi en un castell fer persistent en una base de dades la seva participació i generar estadístiques de les dades desades. Igualment l'aplicació incorpora informació útil del món casteller.

El document explica com s'ha creat l'aplicació des del principi fins al final: introducció al projecte, anàlisi del sistema, disseny tècnic, implementació i conclusions.

És important destacar que el projecte .Castells és un projecte desenvolupat utilitzant l'IDE Xcode d'Apple, el llenguatge de programació Objective-C i l'iOS SDK 5 com a principals eines i que, tot i que pot funcionar a dispositius iPad, està pensat principalment per ser utilitzat sobre dispositius mòbils iPhone d'Apple.

# Taula de continguts

---

## **1. Introducció 8**

---

- 1.1. Justificació i context 8
- 1.2. Objectius del projecte 9
- 1.3. Enfocament i mètode seguit 10
- 1.4. Planificació 12
- 1.5. Maquinari i programari utilitzat 13
- 1.6. Productes obtinguts 14

## **2. Anàlisi del sistema 15**

---

- 2.1. Diagrama de casos d'ús 15
  - 2.1.1. Subsistema d'inici 16
  - 2.1.2. Subsistema de castell 16
  - 2.1.3. Subsistema d'informació 17
  - 2.1.4. Subsistema d'estadístiques 17
- 2.2. Casos d'ús 18
  - 2.2.1. Inici 18
  - 2.2.2. Mostrar les instruccions 18
  - 2.2.3. Sobre .Castells 19
  - 2.2.4. Data - He fet un castell 19
  - 2.2.5. Ciutat 20
  - 2.2.6. Categoria 20
  - 2.2.7. Castell 21
  - 2.2.8. Resultat 22
  - 2.2.9. Lloc 22
  - 2.2.10. Posició 23
  - 2.2.11. Desar 23
  - 2.2.12. Informació 24
  - 2.2.13. Què són els castells? 25
  - 2.2.14. Les colles 25
  - 2.2.15. Detall colla 26
  - 2.2.16. Vocabulari casteller 26
  - 2.2.17. On són les colles? 27
  - 2.2.18. Estadístiques 27
  - 2.2.19. Visualitzar tots els castells 28
  - 2.2.20. Visualitzar resta d'estadístiques 28

### **3. Disseny tècnic 29**

---

- 3.1. Patró de disseny Model - Vista - Controlador 29
- 3.2. Diagrama de classes de Core Data 30
- 3.3. Formes normals a la base de dades 31
- 3.4. Detall de la base de dades 32
- 3.5. Prototip 33

### **4. Implementació 34**

---

- 4.1. Subsistema d'inici 35
- 4.2. Subsistema de castells 35
  - 4.2.18. Recollida de dades 35
  - 4.2.19. Desar les dades 39
- 4.3. Subsistema d'informació 43
- 4.4. Subsistema d'estadístiques 46

### **5. Conclusions 49**

---

- 5.1. Línies de futur 50

### **6. Glosari 52**

---

### **7. Bibliografia 54**

---

# Taula de figures

---

## 1. Introducció 8

---

Figura 1-1. Parts d'un castell	8
Figura 1-2. Exemple d'utilització de Tab Bar amb dos tabs	10
Figura 1-3. Implementació de la pantalla On són les colles? dins el subsistema d'informació	11
Figura 1-4. Diagrama de Gant del projecte	12
Figura 1-5. Exemple de la vista que permet cercar una ciutat	13

## 2. Anàlisi del sistema 15

---

Figura 2-1. Diagrama de casos d'ús general	15
Figura 2-2. Diagrama de casos d'ús del subsistema d'inici	16
Figura 2-3. Diagrama de casos d'ús del subsistema de castell	16
Figura 2-4. Diagrama de casos d'ús del subsistema d'informació	17
Figura 2-5. Diagrama de casos d'ús del subsistema d'estadístiques	17

## 3. Disseny tècnic 29

---

Figura 3-1. Diagrama MVC	30
Figura 3-2. Diagrama de classes Core Dara del projecte .Castells	31
Figura 3-3. Taula ZCASTELL a Core Data	32
Figura 3-4. Frame del vídeo del prototip de l'aplicació	33

## 4. Implementació 34

---

Figura 4-1. Alerta Twitter	34
Figura 4-2. Les tres vistes del subsistema	35
Figura 4-3. Vista He fet un castell, vista de la selecció de colla i la colla seleccionada	36
Figura 4-4. Alerta: no has seleccionat cap colla	36
Figura 4-5. Codi que força el Segue quan es selecciona una fila de la taula filtrada	37
Figura 4-6. Ciutat i exemple de filtre de ciutats	38
Figura 4-7. Vistes Castell diferents	38
Figura 4-8. Vistes posicions diferents	39
Figura 4-9. Vista Desar	40
Figura 4-10. Exemple de swipe to delete	41
Figura 4-11. Detall Tweet colla	41
Figura 4-12. Quatre exemples de Tweets personals amb diferent text	42
Figura 4-13. Alerta: selecciona colla i castell no desat	42
Figura 4-14. Alerta: no tens configurat el compte a Twitter	43
Figura 4-15. Alerta: selecciona colla i Tweet no generat	43

Figura 4-16. Vistes Què són els castells? i Vocabulari casteller.	44
Figura 4-17. Vista Les colles, el detall, generació de Tweet i pàgina web	45
Figura 4-18. Dos exemples de la vista On són les colles?	45
Figura 4-19. Vista principal, les dos vistes de filtratge i els filtres seleccionats	46
Figura 4-20. Les tres estadístiques que es poden generar	47
Figura 4-21. Exportar.	47

## **5. Conclusions 48**

---

Figura 5-1. Comparació vista Informació entre el prototip i la definitiva	48
Figura 5-2. Pantalla principal del blog creat	49

# 1. Introducció

---

## 1.1. Justificació i context

Només a Catalunya hi ha 62 colles castelleres que aglutinen a un mínim de cent persones al seu voltant cada una. Però en determinats moments de la temporada poden arribar a aglutinar moltes més, ja que per fer alguns castells poc habituals com el 3 de 10 es poden arribar a necessitar unes mil persones.

Un castell està dividit en sis parts: pom de dalt, tronc, manilles, folre, pinya i soca. Dins de cada part del castell podem trobar diferents posicions que pot fer un casteller, per exemple, al pom de dalt, podem trobar els dosos, l'aixecador i l'enxaneta.



**Figura 1-1.** Parts d'un castell<sup>1</sup>.

Tot i que fer castells és una feina que només es pot fer en grup, existeixen diferents tasques que impliquen diferents responsabilitats i que fan que la funció i la implicació d'un casteller en l'èxit d'un castell evolucionin amb el temps.

Actualment no existeix cap eina que permeti fer persistent la participació d'un casteller en un castell determinat, en una data determinada i en una població determinada. A la web de la

---

<sup>1</sup> Imatge extreta de la pàgina web: [http://www.castellersdelaviladegracia.org/index.php?option=com\\_content&view=article&id=19&Itemid=48](http://www.castellersdelaviladegracia.org/index.php?option=com_content&view=article&id=19&Itemid=48)



Coordinadora de Colles Castelleres de Catalunya<sup>2</sup> es pot trobar una Base de Dades general de tots els castells fets per una colla, però està lluny d'arribar a nivell unipersonal.

És per aquest motiu que es decideix desenvolupar un recurs que permeti donar solució a aquesta falta d'eines adequades pels castellers.

Paral·lelament es vol afegir a l'aplicació tota la informació possible del món casteller com el vocabulari casteller, la ubicació de les colles al territori i la informació bàsica de les colles.

## 1.2. Objectius del projecte

Així doncs, el projecte té com a principal objectiu crear una aplicació mòbil que permeti a milers de castellers fer persistent les dades d'una actuació personal dins d'una diada castellera. L'aplicació ha de permetre desar la colla a la que pertany el casteller, la ciutat i la data de l'actuació, el castell que s'ha fet, el resultat del mateix i la posició ocupada pel casteller.

Acompanyada a qualsevol base de dades s'ha de crear un apartat d'estadístiques que permetin a l'usuari de l'aplicació cercar en les dades que s'han fet persistents a l'aplicació. No és l'objectiu d'aquest projecte crear moltes estadístiques diferents. És per això que l'aplicació permet a l'usuari poder llistar per una banda tots els castells fets i per l'altra el nom dels castells i els resultats obtinguts en els castells en els que ha participat.

Donat que pel temps de desenvolupament que es tenia no es podia fer una aplicació multiplataforma, es va decidir desenvolupar-la únicament per telèfons iPhone de la companyia Apple. Aquesta decisió es va prendre per la dificultat afegida que representava l'aprenentatge d'un nou llenguatge i entorn de programació. D'aquesta manera, adquirir coneixements en l'entorn de les aplicacions mòbils iOS es convertia en un objectiu del projecte.

Al mateix temps, i donat que el treball final de carrera té una elevada càrrega de treball pràctic, també ha estat un objectiu l'adquisició de les habilitats pràctiques, o experiència, associades a aquests coneixements.

A més a més s'ha pretès crear un producte de màxima qualitat que permeti ser una bona carta de presentació un cop s'acabin els estudis d'Enginyeria Tècnica Informàtica de Gestió. És per això que s'ha d'intentar utilitzar al màxim les eines que proporciona Apple per desenvolupar aplicacions. La cerca de la màxima qualitat i la bona utilització de les eines disponibles són un altre objectiu.

També existeixen uns objectius de caire més personal que no es poden deixar a banda:

Aprendre qüestions relacionades amb l'elaboració d'un projecte informàtic, com ara quina metodologia i planificació s'ha de seguir per gestionar el projecte per tal d'acabar amb èxit.

---

<sup>2</sup> <http://www.cccc.cat/>

Analitzar les funcionalitats que ha de tenir un projecte d'aquestes característiques.

Aprendre a planificar el treball a realitzar en relació al temps disponible per acabar el projecte dins del TFC.

Aprendre tot el que sigui possible d'un llenguatge de programació, fins ara desconegut dins dels estudis, com és Objective-C, d'un entorn de programació nou com és l'IDE Xcode d'Apple i d'un framework que encapsula unes biblioteques desconegudes com és Cocoa Touch.

Posar en pràctica els coneixements adquirits durant els estudis que permetin fer un bon testeig del projecte final.

Generar un codi ben documentat, estructurat i net per tal que el projecte sigui fàcil de mantenir en el futur.

### 1.3. Enfocament i mètode seguit

Xcode, l'IDE per desenvolupar aplicacions iOS, disposa de diferents plantilles que permeten estandarditzar el disseny de les aplicacions. Una d'elles és l'anomenada Tabbed Application. Aquest tipus de plantilla permet crear unes pestanyes que ens porten de manera fàcil a pantalles diferents de l'aplicació. Per desenvolupar aquesta aplicació s'han utilitzat 4 pestanyes o Tab Bar.

D'aquesta manera es va subdividir el projecte en 4 sistemes de funcionalitats diferents: sistema d'inici, sistema de castell, sistema d'informació i sistema d'estadístiques.



**Figura 1-2.** Exemple d'utilització de Tab Bar amb dos tabs<sup>3</sup>.

---

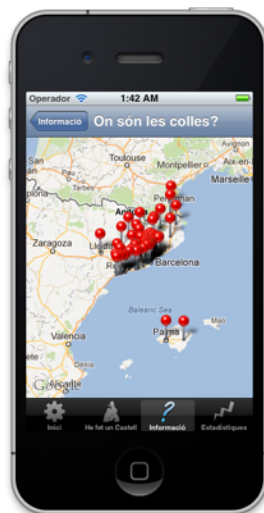
<sup>3</sup> Imatge extreta de la pàgina web: [http://www.techotopia.com/index.php/Creating\\_an\\_iOS\\_5\\_iPhone\\_Multiview\\_Application\\_using\\_the\\_Tab\\_Bar](http://www.techotopia.com/index.php/Creating_an_iOS_5_iPhone_Multiview_Application_using_the_Tab_Bar)

És important tenir en compte que es va decidir utilitzar tot l'entorn de programació recomanat per Apple i que inclou l'IDE Xcode, un maquinari Mac, l'ús d'Objective-C a a llenguatge de programació i la utilització de Cocoa Touch com a framework principal. D'aquesta manera s'ha aconseguit una aplicació totalment nativa per la plataforma seleccionada. D'altra banda es va descartar generar una aplicació a partir d'altres possibles entorns de programació però que poden ser visibles dins d'un iPhone<sup>4</sup>.

Un cop decidit l'entorn i el mètode i un cop generats els documents bàsics que permetien començar a treballar es va procedir a desenvolupar un prototip navegable en el que ja es podia visualitzar l'estructura final de l'aplicació però que no permetia treballar amb les principals funcions del projecte.

La tasca de crear el prototip va ser el primer contacte fort amb l'entorn de programació escollit per aquest projecte i va ser necessari invertir molt esforç. Un cop generat i aprovat el prototip es va donar forma a les funcionalitats principals del projecte implementant la lògica i la persistència.

Un cop superada aquesta fase, potser la de més dificultat del projecte, només quedava emplenar i donar forma a tot el subsistema d'informació. Aquest no era el subsistema més complicat però si que ha requerit moltes hores de feina. Per posar un exemple, la tasca de col·locar totes les colles a un mapa ha requerit cercar les adreces de totes les colles, visualitzar aquestes adreces a GoogleMaps, extreure les coordenades i després traspasar aquestes coordenades a la classe encarregada de col·locar al mapa les marques amb la informació.



**Figura 1-3.** Implementació de la pantalla On són les colles? dins el subsistema d'informació.

---

<sup>4</sup> Per exemple es podria haver desenvolupat una aplicació web que hagués estat visible dins el navegador de l'iPhone.

## 1.4. Planificació

La planificació del projecte s'ha fet tenint en compte les dates d'entrega de les proves d'avaluació continuada i del lliurament final. De fet aquestes entregues intenten determinar l'inici i el final de les diferents fases de desenvolupament d'un projecte informàtic.

D'aquesta manera s'ha dividit la feina per desenvolupar el projecte en quatre fases: pla de treball, anàlisi, disseny i prototip, implementació i entrega. Dins de cada fase s'han intentat temporalitzar les principals tasques que s'han de realitzar. És evident que a la planificació es va realitzar a l'inici del projecte i era molt difícil determinar el temps exacte que es necessitava per, per exemple, desenvolupar un subsistema determinat, així doncs es van determinar les dates d'inici i final de cada subfase com orientatives.

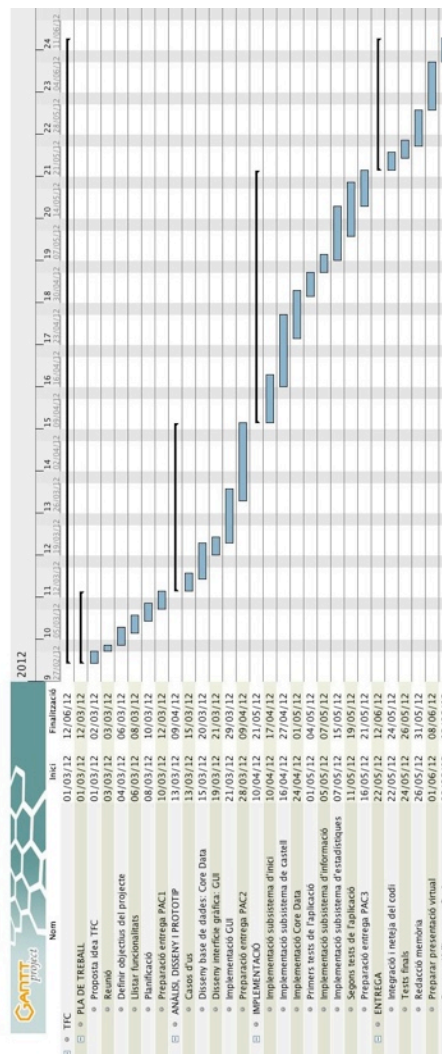
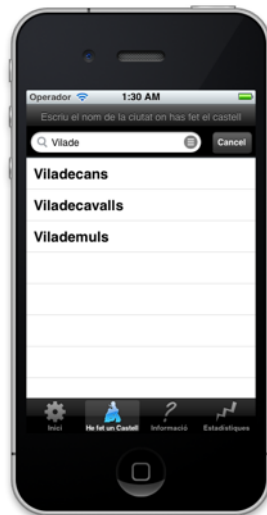


Figura 1-4. Diagrama de Gant del projecte.

Les dates d'inici i final de cada fase del desenvolupament de l'aplicació s'han respectat totalment. En canvi les tasques a realitzar dins de cada fase no han seguit, de manera força general, la planificació inicial. La raó principal és que les tasques la majoria de vegades s'han fet de manera conjunta. Per exemple, en el procés d'implementació s'han implementat subsistemes de manera paral·lela, mentre que a la planificació inicial es reservaven uns dies concrets per cada subsistema.

De la mateixa manera el desconeixement inicial de l'entorn de programació i sobretot del framework Cocoa Touch ha comportat forces traves i estancaments en diferents punts del projecte. Per posar un exemple, un punt crític es pot localitzar quan ja s'apropava la data d'entrega del producte acabat en la programació de la classe que permetia cercar una ciutat de la llista inclosa a una taula. Poder anar a la propera vista un cop es feia una cerca, enviar la ciutat seleccionada a la següent vista i tornar a la vista un cop l'usuari de l'aplicació ja havia deixat enrere aquesta pantalla van representar problemes que van requerir força hores i dies per solucionar-los.



**Figura 1-5.** Exemple de la vista que permet cercar una ciutat.

## 1.5. Maquinari i programari utilitzat

El maquinari utilitzat per desenvolupar el projecte i realitzar tots el documents ha estat un Apple iMac. Tot i que l'entorn de programació iOS porta inclòs un simulador força complet, ha estat necessari poder testar l'aplicació en un iPhone com el que utilitzaran els usuaris finals de l'aplicació. Així doncs, s'ha utilitzat un iPhone 4 per testejar el producte final.

El programari utilitzat és ampli:

- Xcode 4.3.1 com a entorn de desenvolupament integrat (IDE).
- Simulador iOS 5.1 com a simulador integrat al maquinari.
- GanttProject 2.0.10 per fer diagrames de Gantt.

- MagicDraw 16.5 Personal Evaluation com editor UML.
- Pages 4.1 com editor de texts per redactar documents.
- Keynote 5.1.1 per crear la presentació del projecte.
- Adobe Photoshop CS4 per dissenyar les imatges del TabBar.

El llenguatge de programació utilitzat per desenvolupar el projecte ha estat Objective-C, com a frameworks principals Cocoa Touch i Core Data (base de dades) i com a frameworks auxiliars Quartz Core (per personalitzar elements), MessageUI (per enviar correus), Map Kit (per treballar amb mapes) i Twitter (per poder enviar Tweets).

Finalment comentar que ha estat necessari donar-se d'alta al programa Developer d'Apple per poder tenir accés a totes les eines disponibles.

## 1.6. Productes obtinguts

El principal producte obtingut d'aquest TFC és l'aplicació mòbil iOS per iPhone .Castells, d'aquesta manera s'acompleix el principal objectiu del TFC. Aquesta aplicació implementa totes les funcionalitats que es poden veure al capítol ANÀLISIS DEL SISTEMA!!!!!!! . Evidentment al ser la primera aplicació de l'autor el producte resultant no destaca per una interfície gràfica molt espectacular i atractiva. El que sí que s'ha aconseguit és que l'aplicació sigui totalment intuïtiva i sigui molt fàcil la seva utilització.

En les properes setmanes un cop acabada l'assignatura i un cop s'introdueixin alguns petits retocs (petita reestructuració d'alguns botons i introducció d'algunes imatges de fons) l'aplicació s'enviarà a Apple per la seva inclusió a l'App Store<sup>5</sup>. D'aquesta manera es podrà portar a terme l'últim objectiu, i segurament el més important d'aquest TFC, que és aconseguir que l'aplicació estigui disponible per tothom.

El producte obtingut no disposa d'executable i no és un producte que pugui funcionar per si sol. Per fer-lo servir és necessari un iPhone d'Apple, un iPad o alternativament el Simulador iOS disponible als maquinari Mac.

Fer-lo anar amb el simulador no és difícil, tot i que no està totalment operatiu ja que les funcionalitats que utilitzen l'enviament de correu electrònic no funcionen del tot amb el simulador. De fet, el simulador fa veure que envia el correu però en realitat mai l'envia ja que l'aplicació Mail no està configurada al simulador i no es pot configurar.

Fer anar l'aplicació en un iPhone o un iPad és possible sempre i quan es demani expressament al desenvolupador que es vol testejar l'aplicació. Per instal·lar .Castells és pot utilitzar l'eina que ofereix TestFlight<sup>6</sup> i que permet descarregar l'aplicació.

---

<sup>5</sup> Botiga d'aplicacions d'Apple

<sup>6</sup> <https://www.testflightapp.com/>

Altres productes complementaris han estat els textos parcials, el vídeo presentació i aquesta memòria.

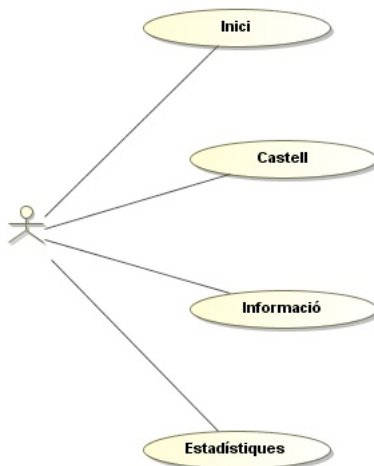
## 2. Anàlisi del sistema

---

### 2.1. Diagrama de casos d'ús

Com s'ha explicat anteriorment, el projecte està dividit en quatre subsistemes diferents: subsistema d'inici, subsistema de castell, subsistema d'informació i subsistema d'estadístiques.

.Castells és una aplicació senzilla i amb uns fluxos d'utilització molt concrets. No s'assembla en res a una aplicació per un ordinador de sobretaula que sol incorporar multitud d'opcions per l'usuari. A més s'ha de tenir en compte que el típic usuari encarregat d'administrar el sistema perd tot el sentit en una aplicació mòbil i sobretot, en una aplicació simple com aquesta. Per tot això .Castells només incorpora un únic actor que pot interactuar amb l'aplicació. En el següent diagrama s'observa com ho fa amb els quatre subsistemes.



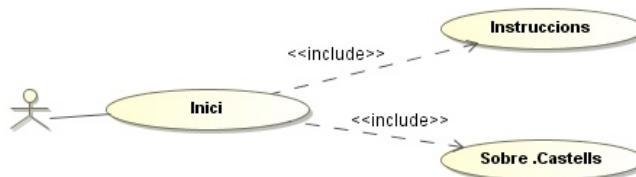
**Figura 2-1.** Diagrama de casos d'ús general.

La divisió en subsistemes permet dividir el projecte en quatre paquets. Així doncs els següents diagrames representen els casos d'us de cada un dels subsistemes.

#### 2.1.1. Subsistema d'inici

---

A continuació el diagrama de casos d'ús del subsistema d'inici.

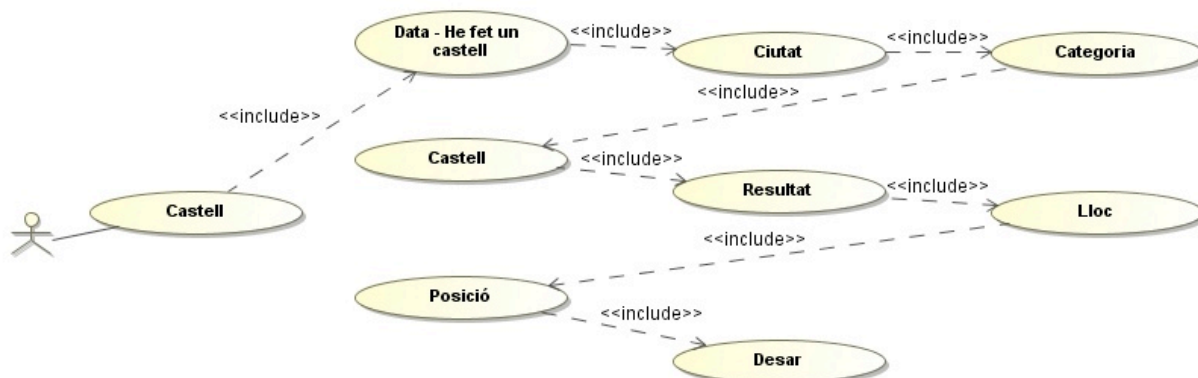


**Figura 2-2.** Diagrama de casos d'ús del subsistema d'inici.

### 2.1.2. Subsistema de castell

---

A continuació el diagrama de casos d'ús del subsistema de castell.



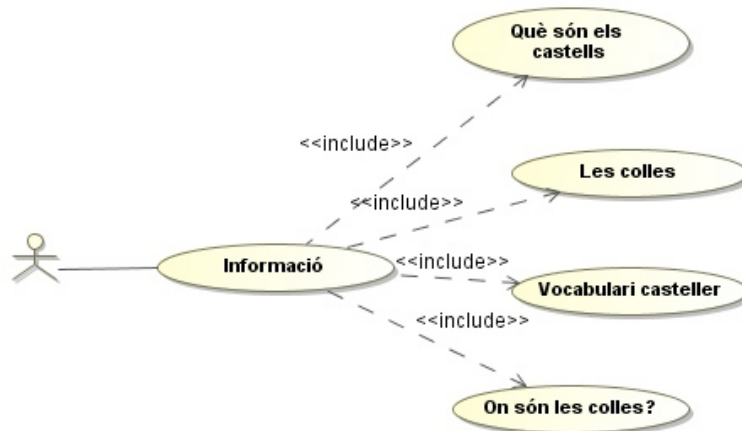
**Figura 2-3.** Diagrama de casos d'ús del subsistema de castell.

### 2.1.3. Subsistema d'informació

---

A continuació el diagrama de casos d'ús del subsistema d'informació.



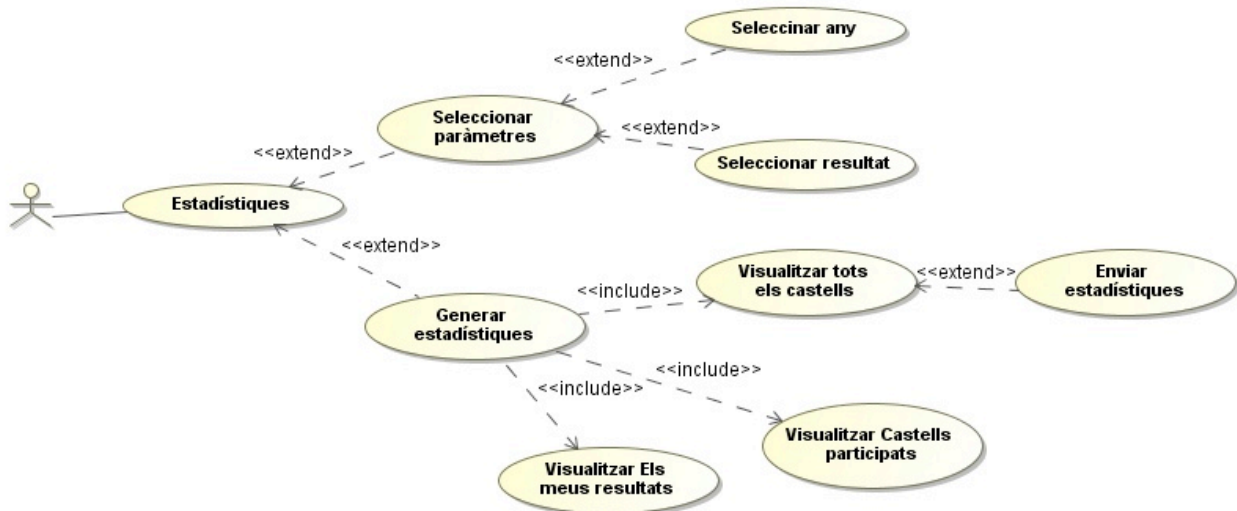


**Figura 2-4.** Diagrama de casos d'ús del subsistema d'informació.

### 2.1.4. Subsistema d'estadístiques

---

A continuació el diagrama de casos d'ús del subsistema d'estadístiques.



**Figura 2-5.** Diagrama de casos d'ús del subsistema d'estadístiques.

## 2.2. Casos d'ús

A continuació es presenten els casos d'us detallats de l'aplicació

### 2.2.1. Inici

---

Objectiu	Vista que permet anar a les vistes inicials de l'aplicació.
Actors implicats	Usuari
Casos d'us relacionats	* Instruccions * Sobre.Castells
Precondició	Cap.
Postcondició	Cap.
Flux normal	* L'usuari selecciona un dels botons. * S'obre una nova vista
Excepcions	Cap.
Altres observacions	Cap.

### 2.2.2. Mostrar les instruccions

---

Objectiu	Permet obrir una vista on s'expliquen les instruccions de l'aplicació
Actors implicats	Usuari.
Casos d'us relacionats	Cap
Precondició	Cap
Postcondició	Cap
Flux normal	* L'usuari clica al botó instruccions. * S'obre una nova vista amb les instruccions detallades de l'aplicació.
Excepcions	Cap
Altres observacions	Es pot tornar a la vista anterior amb el botó que apareix al controlador de navegació de l'aplicació.

### 2.2.3. Sobre .Castells?

---

Objectiu	Permet obrir una vista amb les dades bàsiques de l'aplicació. Que és .Castells, versió, qui ha fet l'aplicació i com contactar amb el desenvolupador.
Actors implicats	Usuari
Casos d'us relacionats	Cap.
Precondició	* L'usuari té configurat un compte de Twitter al seu dispositiu mòbil. * L'usuari té configurat un compte de Mail al seu dispositiu mòbil * El dispositiu mòbil te connexió a Internet disponible.
Postcondició	Cap.
Flux normal	* L'usuari clica en el botó <i>Qui ha fet això?</i> . * S'obre una nova vista amb les dades. * Si es vol enviar un Tweet al desenvolupador es clica a la icona de Twitter. * Si el vol enviar un correu al desenvolupador es clica a la icona de Mail.
Excepcions	Si l'usuari vol enviar un Tweet i no te configurat cap compte de Twitter apareix una alerta.
Altres observacions	Es pot tornar a la vista anterior amb el botó que apareix al controlador de navegació de l'aplicació.

#### 2.2.4. Data - He fet un castell

Objectiu	Permet començar la selecció de l'activitat castellerà en la que ha participat l'usuari de l'aplicació.
Actors implicats	Usuari
Casos d'us relacionats	* Ciutat
Precondició	Cap
Postcondició	Cap
Flux normal	* L'usuari selecciona una colla castellerà. * L'usuari selecciona una data. * L'usuari clica al botó <i>He fet un castell</i> . * Es visualitza la propera vista.
Excepcions	Si l'usuari no selecciona cap colla apareix una alerta.

Altres observacions	<ul style="list-style-type: none"> <li>* La data i la colla castellera de l'usuari queden enregistrades per poder afegir les dades a la base de dades un cop finalitzi la selecció de dades a fer persistents.</li> <li>* El selector de dates no permet seleccionar una data superior al dia actual.</li> </ul>
---------------------	--

### 2.2.5. Ciutat

Objectiu	Permet seleccionar la ciutat on s'ha fet el castell.
Actors implicats	Usuari
Casos d'us relacionats	<ul style="list-style-type: none"> <li>* Data - He fet un castell</li> <li>* Categoria</li> </ul>
Precondició	Cap.
Postcondició	Cap.
Flux normal	<ul style="list-style-type: none"> <li>* L'usuari selecciona una ciutat de la taula o alternativament,</li> <li>* Escriu el nom de la ciutat on ha fet el castell al diàleg de cerca i apareix una llista de les ciutats que coincideixen amb les dades introduïdes per l'usuari i finalment selecciona una ciutat de la taula proporcionada.</li> <li>* Es visualitza la propera vista.</li> </ul>
Excepcions	Cap.
Altres observacions	<ul style="list-style-type: none"> <li>* La ciutat on ha fet el castell l'usuari queda enregistrada per poder afegir les dades a la base de dades un cop finalitzi la selecció de dades a fer persistents.</li> <li>* Es pot tornar a la vista anterior amb el botó que apareix al controlador de navegació de l'aplicació.</li> </ul>

### 2.2.6. Categoria

Objectiu	Permet seleccionar la categoria del castell que s'ha fet
Actors implicats	Usuari
Casos d'us relacionats	<ul style="list-style-type: none"> <li>* Ciutat</li> <li>* Castell</li> </ul>
Precondició	Cap.

Postcondició	Cap.
Flux normal	* L'usuari selecciona una categoria de la taula proporcionada. * Es visualitza la propera vista.
Excepcions	Cap.
Altres observacions	* Es pot tornar a la vista anterior amb el botó que apareix al controlador de navegació de l'aplicació.

### 2.2.7. Castell

---

Objectiu	Permet seleccionar el castell que ha fet l'usuari.
Actors implicats	Usuari
Casos d'us relacionats	* Categoria. * Resultat
Precondició	Cap.
Postcondició	Cap.
Flux normal	* Depenent de la categoria que s'ha seleccionat en la vista anterior apareixen uns castells diferents arregats a la taula de la vista. * L'usuari selecciona una castell a la taula proporcionada. * Es visualitza la propera vista.
Excepcions	Cap.
Altres observacions	* El castell que ha fet l'usuari queda enregistrada per poder afegir les dades a la base de dades un cop finalitzi la selecció de dades a fer persistents. * Es pot tornar a la vista anterior amb el botó que apareix al controlador de navegació de l'aplicació.

### 2.2.8. Resultat

---

Objectiu	Permet seleccionar el resultat del castell
----------	--

Actors implicats	Usuari
Casos d'us relacionats	* Castell. * Lloc.
Precondició	Cap.
Postcondició	Cap.
Flux normal	* L'usuari selecciona un resultat de la taula proporcionada. * Es visualitza la propera vista
Excepcions	Cap.
Altres observacions	* El resultat del castell que ha fet l'usuari queda enregistrat per poder afegir les dades a la base de dades un cop finalitzi la selecció de dades a fer persistents. * Es pot tornar a la vista anterior amb el botó que apareix al controlador de navegació de l'aplicació.

### 2.2.9. Lloc

---

Objectiu	Permet seleccionar el lloc o la posició general en la que ha participat l'usuari.
Actors implicats	Usuari.
Casos d'us relacionats	* Resultat. * Posició.
Precondició	Cap.
Postcondició	Cap.
Flux normal	* L'usuari selecciona una lloc o posició general de la taula proporcionada. * Es visualitza la propera vista.
Excepcions	Cap.
Altres observacions	* El lloc o posició general que l'usuari tenia al castell queda enregistrada per poder afegir les dades a la base de dades un cop finalitzi la selecció de dades a fer persistents. * Es pot tornar a la vista anterior amb el botó que apareix al controlador de navegació de l'aplicació.

### 2.2.10. Posició

---

Objectiu	Permet seleccionar la posició concreta en la que ha participat l'usuari.
Actors implicats	Usuari.
Casos d'us relacionats	* Lloc. * Desar.
Precondició	Cap.
Postcondició	Cap.
Flux normal	* Depenent del lloc que s'ha seleccionat en la vista anterior apareixen unes posicions diferents carregades a la taula de la vista. * L'usuari selecciona un lloc concret de la taula proporcionada. * Es visualitza la propera vista.
Excepcions	Cap.
Altres observacions	* La posició concreta que l'usuari tenia al castell queda enregistrada per poder afegir les dades a la base de dades un cop finalitzi la selecció de dades a fer persistents. * Es pot tornar a la vista anterior amb el botó que apareix al controlador de navegació de l'aplicació.

### 2.2.11. Desar<sup>7</sup>

Objectiu	Permet afegir a la base de dades de l'aplicació les dades que s'han seleccionat.
Actors implicats	Usuari.
Casos d'us relacionats	* Data - He fet un castell * Ciutat * Castell * Resultat * Lloc * Posició
Precondició	Cap.
Postcondició	Les dades queden desades a la Base de dades

<sup>7</sup> swipe to delete consisteix en moure el dit per sobre d'una fila de dreta a esquerra.

Flux normal	<ul style="list-style-type: none"> <li>* Es mostra un detall de les dades seleccionades i que es volen fer persistents.</li> <li>* Si l'usuari selecciona el botó Desar, les dades es guarden a la base de dades i es mostra una vista amb tots els castells desats fins el moment ordenats per data.</li> <li>* Si l'usuari selecciona el botó Desar + Tweet colla, les dades es guarden a la base de dades i es mostra una vista amb tots els castells desats fins al moment ordenats per data. A més a més es genera un tweet general i es visualitza en la nova vista perquè pugui ser piulat.</li> <li>* Si l'usuari selecciona el botó Generar Tweet personal, es visualitza una vista amb tots els castells desats fins el moment ordenats per data i es genera un tweet personal i es visualitza en la nova vista perquè pugui ser piulat.</li> <li>* Un cop visualitzada la llista amb tots els castells desats a la base de dades és possible esborrar-los un a un fent un swipe to delete.</li> </ul>
Excepcions	<ul style="list-style-type: none"> <li>* Si l'usuari intenta desar un castell i no va seleccionar cap coll a apareix una alerta i el castell no es desa.</li> <li>* Si l'usuari intenta generar un Tweet i no te configurat el compte de Twitter apareix una alerta.</li> <li>* Si l'usuari intenta generar un Tweet personal i no ha seleccionat cap colla apareix una alerta i el Tweet no es genera.</li> </ul>
Altres observacions	<p>Seleccionant el Tab Bar inferior d'aquest subsistema és possible tornar a la primera pantalla.</p>

### 2.2.12. Informació

Objectiu	Vista que permet anar a les vistes amb informació del món casteller.
Actors implicats	Usuari
Casos d'us relacionats	<ul style="list-style-type: none"> <li>* Què són els castells?</li> <li>* Les colles</li> <li>* Vocabulari casteller</li> <li>* On són les colles?</li> </ul>
Precondició	Cap.
Postcondició	Cap.
Flux normal	<ul style="list-style-type: none"> <li>* L'usuari selecciona un dels botons.</li> <li>* S'obre una nova vista</li> </ul>
Excepcions	Cap.



Altres observacions	Cap.
---------------------	------

### 2.2.13. Què són els castells?

---

Objectiu	Permet obrir una vista on hi ha una explicació de què són els castells.
Actors implicats	Usuari
Casos d'us relacionats	* Informació.
Precondició	Cap.
Postcondició	Cap.
Flux normal	* L'usuari clica en el botó <i>Què són els castells?</i> * S'obre una nova vista amb les dades.
Excepcions	Cap
Altres observacions	Es pot tornar a la vista anterior amb el botó que apareix al controlador de navegació de l'aplicació.

### 2.2.14. Les colles

---

Objectiu	Permet accedir a una vista on tenim els escuts de totes les colles per poder accedir a tota la informació de les mateixes.
Actors implicats	Usuari
Casos d'us relacionats	* Informació.
Precondició	Cap.
Postcondició	Cap.
Flux normal	* L'usuari clica sobre l'escut d'una colla. * S'obre una vista amb la informació detallada de la colla.
Excepcions	Cap.
Altres observacions	Es pot tornar a la vista anterior amb el botó que apareix al controlador de navegació de l'aplicació.

### 2.2.15. Detall colla

Objectiu	Permet accedir a la informació de la colla que s'ha seleccionat.
Actors implicats	Usuari
Casos d'us relacionats	* Les colles.
Precondició	Cap.
Postcondició	Cap.
Flux normal	<p>* L'usuari pot llegir els detalls (localitat, adreça, color camisa, any fundació, millor castell, millor actuació, diades, pàgina web i Twiter) de la colla que ha seleccionat.</p> <p>* Si l'usuari clica sobre la pàgina webde la colla, s'obre el navegador nadiu de l'iPhone amb la pàgina web de la colla castellera.</p> <p>* Si l'usuari clica sobre el Twitter de la colla s'obre el diàleg nadiu de Twitter i pot enviar un tweet a la colla.</p>
Excepcions	Si l'usuari de l'aplicació intenta enviar un Tweet i no te configurat cap compte de Twitter al seu dispositiu mòbil apareix una alerta.
Altres observacions	Es pot tornar a la vista anterior amb el botó que apareix al controlador de navegació de l'aplicació.

### 2.2.16. Vocabulari casteller

Objectiu	Permet obrir una vista on hi ha vocabulari casteller.
Actors implicats	Usuari
Casos d'us relacionats	* Informació.
Precondició	Cap.
Postcondició	Cap.
Flux normal	<p>* L'usuari clica en el botó <i>Vocabulari casteller</i>.</p> <p>* S'obre una nova vista amb les dades.</p>
Excepcions	Cap
Altres observacions	Es pot tornar a la vista anterior amb el botó que apareix al controlador de navegació de l'aplicació.

### 2.2.17. On són les colles?

Objectiu	Permet obrir una vista d'un mapa on es poden visualitzar marcades totes les colles castelleres existents.
Actors implicats	Usuari
Casos d'us relacionats	* Informació.
Precondició	* El dispositiu mòbil te connexió a Internet disponible.
Postcondició	Cap.
Flux normal	* L'usuari clica en el botó <i>On són les colles?</i> . * S'obre la vista amb el mapa
Excepcions	Cap
Altres observacions	Es pot tornar a la vista anterior amb el botó que apareix al controlador de navegació de l'aplicació.

### 2.2.18. Estadístiques

Objectiu	Vista que permet seleccionar paràmetres i quines estadístiques volem generar.
Actors implicats	Usuari
Casos d'us relacionats	* Selecció paràmetre: any * Selecció paràmetre: resultat * Estadística: castells participats * Estadística: les meves posicions
Precondició	Cap.
Postcondició	Cap.
Flux normal	* L'usuari selecciona un, dos o cap paràmetre per filtrar les estadístiques. * L'usuari selecciona una de les tres estadístiques implementades. * S'obre una nova vista.
Excepcions	Cap.
Altres observacions	Cap.

### 2.2.19. Visualitzar tots els castells

Objectiu	Permet visualitzar totes els castells desats a la base de dades i exportar les dades via correu electrònic.
Actors implicats	Usuari
Casos d'us relacionats	* Estadístiques
Precondició	Cap.
Postcondició	Cap.
Flux normal	<ul style="list-style-type: none"> <li>* Es visualitzen tots els castells desats a la base de dades ordenats per data.</li> <li>* Si l'usuari clica al botó Exportar situat al controlador de navegació s'obre una vista que permet enviar un correu electrònic amb tots els castells desats a la base de dades en forma de taula HTML.</li> <li>* Tamé és possible esborrar castells des d'aquesta pantalla utilitzant el swipe to delete.</li> </ul>
Excepcions	Cap.
Altres observacions	* Es pot tornar a la vista anterior amb el botó que apareix al controlador de navegació de l'aplicació.

### 2.2.20. Visualitzar resta d'estadístiques

Objectiu	Permet seleccionar el resultat dels castells sobre els que es vol generar l'estadística.
Actors implicats	Usuari
Casos d'us relacionats	* Estadístiques
Precondició	Cap.
Postcondició	Cap.
Flux normal	<ul style="list-style-type: none"> <li>* Si l'usuari selecciona el botó Castells participats s'obre una vista amb una llista amb el número de castells participats de cada tipus.</li> <li>* Si l'usuari selecciona el botó Els meus resultats s'obre una vista amb una llista amb el número de resultats de cada tipus en els que ha participat l'usuari.</li> </ul>

Excepcions	Cap.
Altres observacions	<ul style="list-style-type: none"> <li>* Es pot tornar a la vista anterior amb el botó que apareix al controlador de navegació de l'aplicació.</li> <li>* El resultat seleccionat queda enregistrat per poder fer la crida a la base de dades.</li> </ul>

### 3. Disseny tècnic

#### 3.1. Patró de disseny Model - Vista - Controlador

En el desenvolupament de programari existeixen diferents patrons de disseny que permeten trobar diferents solucions als problemes que es troben els desenvolupadors. Els patrons són una forma coneguda d'organitzar els dissenys que permet recollir l'experiència adquirida.

En entorn de programació iOS, el patró de disseny recomanat per Apple és el patró Model – Vista – Controlador o MVC<sup>8</sup>. Aquest patró de disseny separa de manera clara les classes de negoci i de dades (Model), la interfície gràfica que es mostra a l'usuari (Vista) i els controladors que les enllacen (Controlador). Cap objecte pot ser dos coses a la vegada.

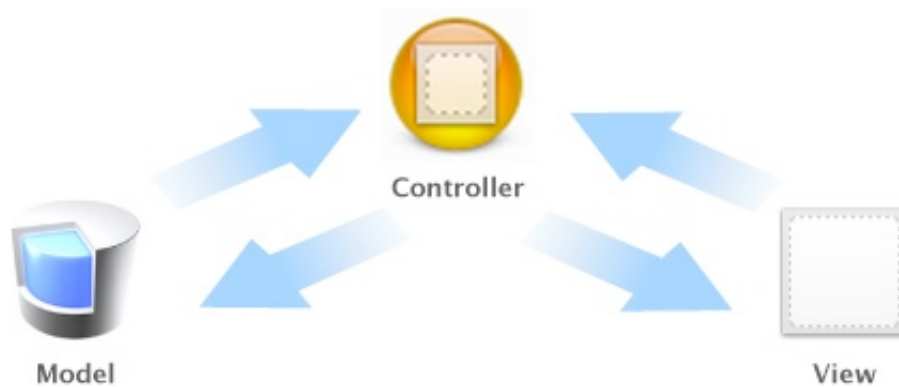
Així doncs, totes les classes es poden classificar dins el projecte segons siguin d'un tipus o un altre. Per traduir una mica el patró a un llenguatge que s'ha estudiat amb anterioritat es pot dir que les classes Model són les que es coneixen com classes de negoci i són les classes relacionades amb la base de dades.

Les classes Vista són les que es coneixen com classes frontera. Són les encarregades de mostrar la interfície gràfica de l'aplicació a l'usuari. Mai una classe vista és la responsable d'emmagatzemar dades, només ha de ser responsable de mostrar-les.

Finalment les classes Controlador són les que es poden identificar com classes de control. Són les encarregades de posar en contacte la vista i el model. De fet són les classes que solen aglutinar una major part de la lògica de l'aplicació. Els controladors poden comunicar-se a les vistes i als models, però els models i les vistes no s'han de conèixer mai.

Aquest tipus de model es pot visualitzar millor amb el següent esquema:

<sup>8</sup> Model–View–Controller en la versió anglesa original.



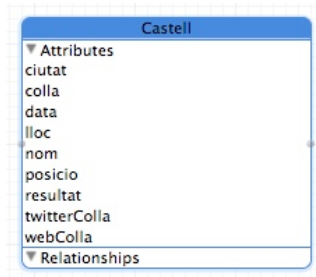
**Figura 3-1.** Diagrama MVC<sup>9</sup>.

### 3.2. Diagrama de classes de Core Data

Per una aplicació creada amb Xcode i utilitzant el framework Core Data, la semblança entre les classes de negoci i les taules de la base de dades és molt gran, quasi es pot dir que una i l'altra són equivalents.

Desenvolupant la base de dades amb Core Data s'han de definir una serie d'entitats que equivalen a les classes de negoci. També, igual que en una classe, és necessari crear, dins de cada entitat, uns atributs. Finalment les diferents entitats es relacionen entre elles per mitjà de relacions.

La següent figura mostra el diagrama d'entitats que és necessari per desar les dades de l'aplicació que s'està desenvolupant. Com es pot observar està a mig camí entre un diagrama de classes UML clàssic i un diagrama ER per una base de dades.



**Figura 3-2.** Diagrama de classes Core Dara del projecte .Castells.

<sup>9</sup> Diagrama estret de la documentació oficial Apple.

De les dades que s'han recollit es vol guardar la ciutat de l'actuació, la colla castellera, la data en la que s'ha fet el castell, el lloc i la posició de l'usuari, el castell fet i el resultat. A més a més, a partir de la colla castellera seleccionada, la classe encarregada de desar el castell te implementada una funció que permet afegir al castell que es vol desar el Twitter i la web de la colla.

De totes aquestes dades desades a la base de dades ara mateix, la web de la colla no s'utilitza en cap context de l'aplicació.

### 3.3. Formes normals a la base de dades

La teoria de la normalització pretén, com a principi bàsic, que tota relació descriu un concepte semàntic únic. Així doncs la teoria de la normalització permet reconèixer els casos en els quals aquest principi no es compleix. El mecanisme per fer-ho són les formes normals (FN), així doncs una relació està en una determinada FN si satisfà un conjunt de restriccions determinades.

A major nivell de forma normal menor és el risc d'inconsistències i repeticions. A canvi la informació desada en la base de dades és de més difícil accés i, sobretot, d'accés més lent. Les formes normals més habituals solen estar entre la 3FN i la 4FN. Aquestes són les FN recomanades quan es treballa amb gestors de bases de dades basats amb servidor. Però s'ha de tenir en compte que la potència del processador d'un dispositiu mòbil no és la mateixa que la d'un ordinador de taula i que el gestor de bases de dades que fa servir Core Data<sup>10</sup> no és pot comparar en cap cas als gestors comercials ni en la seva potència ni en la seva versatilitat.

Per tot això és recomanable no complicar molt la base de dades i, segurament renunciar a algun nivell de forma normal per tal d'agilitzar al màxim la utilització del processador del dispositiu mòbil. Per tant, tenir una base de dades entre 2FN i 3FN s'ha de considerar correcte en dispositius mòbils i amb bases de dades poc complexes.

D'aquesta manera la base de dades que es fa servir a .Castells es extremadament senzilla i compleix sense problemes la 2FN. Per respectar la 3FN necessitaria una nova entitat colla on s'hauria d'incloure les dades relacionades directament amb la colla: Twitter i web.

Tot i que implementar la base de dades en 3FN no necessàriament afectaria el rendiment de l'aplicació, si que complicaria la posada en marxa de l'aplicació per la necessitat de precarregar tota la informació relativa a les colles en la base de dades i no representaria cap millora real a l'hora d'accedir a la base de dades.

### 3.4. Detall de la base de dades

---

<sup>10</sup> Core Data utilitza SQLite.

Fins ara només s'ha pogut veure el diagrama d'entitats que crea Core Data, però el detall de la entitat Castell és la que es mostra a la següent taula:

Atribut	Tipus	Descripció
id	Numèric	Identificador únic per cada castell.
ciutat	String	Nom de la ciutat on s'ha fet el castell.
colla	String	Nom de la colla amb la que s'ha fet el castell.
data	Date	Data en la que s'ha fet el castell.
lloc	String	Lloc general que s'ocupava quan s'ha fet el castell.
nom	String	Nom del castell fet.
posicio	String	Posició concreta que s'ocupava quan s'ha fet el castell.
resultat	String	Resultat del castell.
twitterColla	String	Twitter de la colla castellera.
webColla	String	Web de la colla castellera.

**Figura 3-3.** Taula ZCASTELL a Core Data

Si s'observa el diagrama d'entitats presentat anteriorment es pot observar que l'id no apareix. Això és degut a que és Core Data qui s'encarrega de crear i gestionar l'identificador únic de les taules.

La inclusió de l'id per part de Core Data es pot comprovar si s'obre l'arxiu CastellsDB.sqlite que és el document gestor de la base de dades. Aquest document és pot trobar fàcilment a les carpetes d'execució del Simulador iOS, donat que és quan s'executa l'aplicació que es crea la base de dades. Aquesta és la definició de taula que es pot trobar:

- CREATE TABLE ZCASTELL ( Z\_PK INTEGER PRIMARY KEY, Z\_ENT INTEGER, Z\_OPT INTEGER, ZDATA TIMESTAMP, ZCIUTAT VARCHAR, ZCOLLA VARCHAR, ZLLOC VARCHAR, ZNOM VARCHAR, ZPOSICIO VARCHAR, ZRESULTAT VARCHAR, ZTWITTERCOLLA VARCHAR, ZWEBCOLLA VARCHAR )

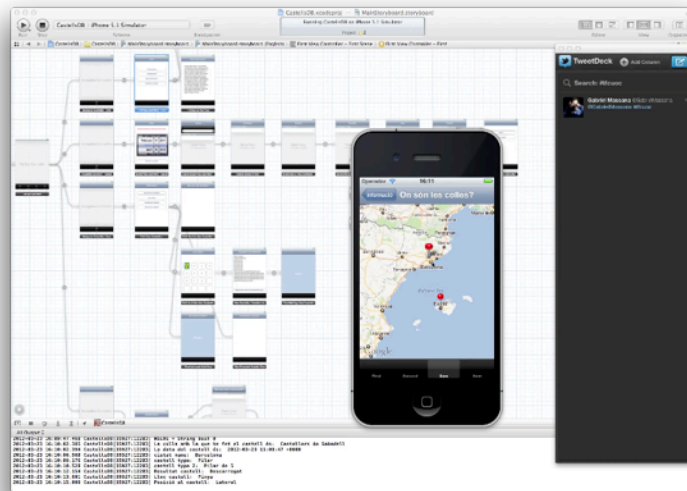


És important destacar que aquest detall de la creació de taules està totalment gestionat per Core Data i que el programador que utilitza el framework no l'ha d'implementar en cap moment.

### 3.5. Prototip

Durant la realització del projecte, com ja s'ha comentat amb anterioritat, es va desenvolupar un prototip el qual es pot visualitzar en el vídeo que es pot descarregar en el següent enllaç:

- <http://uploaded.to/file/eticf1aw>



**Figura 3-4.** Frame del vídeo del prototip de l'aplicació.

S'ha de tenir en compte que el prototip no incorpora alguns casos d'ús i pantalles finalment inclosos en el producte final.

## 4. Implementació

### 4.1. Subsistema d'inici

Aquest subsistema és el més senzill de l'aplicació i ha de ser eliminat abans de la comercialització de l'aplicació. Si s'observa el vídeo del prototip que es presentava en l'anterior apartat es pot comprovar com aquest subsistema ha perdut la meitat dels botons que tenia.

Ara mateix està compost de tres úniques vistes. La primera vista<sup>11</sup> i que fa de pantalla d'inici del subsistema està feta a partir de cel·les estàtiques d'una taula que fan la funció de botons. De fet, tot i que no ho sembli en molts casos, la pràctica totalitat de l'aplicació està implementada a sobre de taules.

Un bon exemple d'això últim són les altres dues vistes del subsistema, tant la vista Instruccions com la vista Sobre .Castells estan fetes sobre taules personalitzades. Per fer-ho ha estat necessari incloure requadres de text<sup>12</sup> a les files i modificar l'aparença. D'aquesta manera s'han modificat els colors, el contorn dels requadres i l'ombra.

A més Sobre .Castells incorpora la possibilitat d'enviar un tweet i un correu electrònic al desenvolupador. És important tenir en compte que la opció d'enviar un tweet al desenvolupador era un cas d'us que incorporava un botó a la pantalla d'inici del subsistema que finalment va ser eliminada. Si l'usuari de l'aplicació intenta enviar un Tweet i no té configurat cap compte de Twitter, s'ha creat una alerta per avisar-lo del problema.



**Figura 4-1.** Alerta Twitter.

Finalment comentar que un cop s'elimini el subsistema, una de les vistes serà recol·locada en un altre subsistema i l'altre serà eliminada. Concretament la vista Instruccions ha de ser eliminada donat que una aplicació senzilla i fàcil de navegar no necessita instruccions. Sobre .Castells serà recol·locada al subsistema de castell i s'afegirà un botó a la barra de navegació de la primera vista del subsistema.

---

<sup>11</sup> Les vistes són objectes anomenats UIView

<sup>12</sup> Els requadres de text són objectes anomenats UITextView



Figura 4-2. Les tres vistes del subsistema.

## 4.2. Subsistema de castells

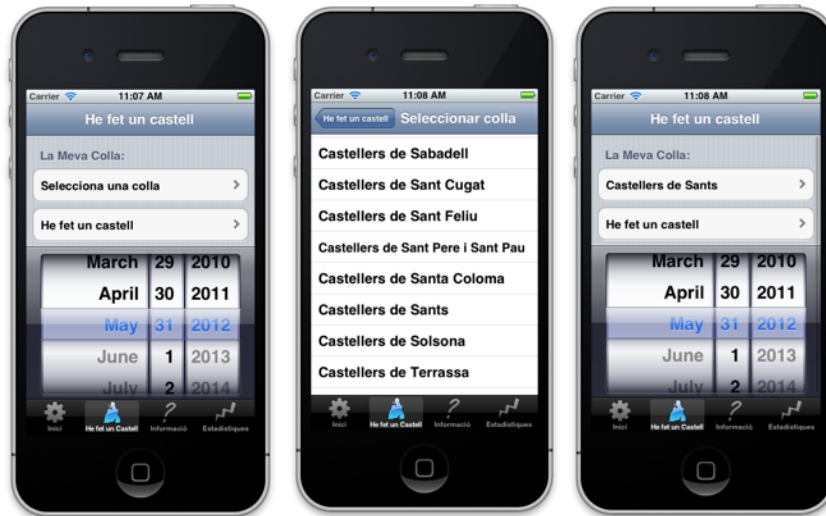
Aquest subsistema té dos parts clarament diferenciades i que, de fet, han estat treballades totalment separades durant la implementació de l'aplicació. La primera part és la recollida de dades del castell fet per l'usuari, la segona és l'encarregada d'emmagatzemar les dades recollides i generar tweets.

### 4.2.1. Recollida de dades

---

La selecció de les dades relatives al castell fet es fa a partir de set vistes creades sobre taules. Cada cop que l'usuari selecciona una dada automàticament passa a la següent taula i així fins que apareix la possibilitat de desar les dades.

Donat que la selecció de colla es volia incloure en la primera vista del subsistema i no es podia afegir una taula sencera, s'ha creat una estructura de vistes delegades que fa que quan es vol seleccionar una colla es vagi a una segona vista on apareix una llista amb totes les colles actuals, un cop es selecciona una colla de la taula es torna automàticament a carregar la primera vista.



**Figura 4-3.** Vista He fet un castell, vista de la selecció de colla i la colla seleccionada.

Com que la selecció de colla és necessària, si l'usuari no selecciona cap colla castellerà apareix una alerta avisant que ha de seleccionar-la.



**Figura 4-4.** Alerta: no has seleccionat cap colla.

Una part complicada d'aquesta part del subsistema ha estat la vista on es pot seleccionar la ciutat on s'ha fet el castell. Donat que la llista és llarga, era necessari implementar algun tipus de filtratge.

Per fer-ho s'utilitza la barra de cerca<sup>13</sup> per filtra les ciutats i pobles llistades a la taula. La cerca no és sensible a majúscules i minúscules i permet trobar el text filtrat en qualsevol posició dins el nom del poble o ciutat.

S'han inclòs en la taula quasi 400 pobles i ciutats. S'ha decidit agrupar-los per ciutats amb colles castelleres, ciutats amb diades i finalment altres ciutats.

Per fer la selecció de ciutats a afegir a la taula s'han cercat les ciutats on han actuat les colles castelleres a la pàgina web de la Coordinadora de Colles Castelleres de Catalunya.

<sup>13</sup> Les barres de cerca són objectes anomenats UISearchBar

Tot i que el nombre de ciutats afegides és gran, és evident que en properes versions pot ser una bona idea afegir la totalitat de ciutats i pobles dels Països Catalans.

Quan es selecciona una ciutat es passa a la següent pantalla. La data, la colla castellera i la ciutat seleccionada passen a la següent pantalla per poder ser emmagatzemats al final del procés.

Com ja es va comentar en capítols anteriors el filtratge de les ciutats va representar un punt crític en la implementació de l'aplicació. El problema que va aparèixer implementant la cerca va ser que un cop es filtraven les ciutats no es podia anar a la següent pantalla.

Amb l'aparició de l'iOS 5 SDK van aparèixer els Storyboards, aquests permeten connectar de manera gràfica totes les classes frontera de la nostra aplicació i faciliten molt la implementació de codi. Ara, de fet, és l'IDE l'encarregat de gestionar quasi de manera exclusiva les classes frontera i la connexió entre elles.

El problema a l'hora d'implementar la vista Ciutat va aparèixer un cop feta la cerca donat que la gestió automàtica de transició entre vistes<sup>14</sup> no es feia. Per solucionar-ho va ser necessari forçar a la vista a fer la transició entre vistes un cop s'havia fet una cerca, a continuació el codi implementat per solucionar el problema:

```
- (void)tableView:(UITableView *)tableView didSelectRowAtIndexPath:(NSIndexPath *)indexPath
{
    if (self.searchDisplayController.isActive) {
        NSMutableArray *mutabllarray = [searchData objectAtIndex:indexPath.section];
        ciutat = [mutabllarray objectAtIndex:indexPath.row];
        [self performSegueWithIdentifier:@"categoria" sender:self];
        searchBar.text = @"";
    }
}
```

**Figura 4-5.** Codi que força el Segue quan es selecciona una fila de la taula filtrada.

<sup>14</sup> En programació iOS aquesta transició s'anomena Segue.



**Figura 4-6.** Ciutat i exemple de filtre de ciutats.

Quan es selecciona una categoria o un lloc aquestes dades permeten carregar llistes diferents a les vistes que hi ha a continuació. D'aquesta manera depenent de quina categoria triem es carrega una llista de castells diferent.



**Figura 4-7.** Vistes Castell diferents.

I depenent de quin lloc es selecciona també es carreguen unes dades diferents a la següent pantalla:



**Figura 4-8.** Vistes posicions diferents.

#### 4.2.2. Desar les dades

---

Al igual que la resta de vistes que implementen botons, la vista principal que permet desar les dades està composta per cel·les estàtiques d'una taula.

En la part superior es mostra un detall del castell que es vol desar a la base de dades. Per col·locar tota la informació en una cel·la d'una taula ha estat necessari personalitzar-la i incloure tres tires de text<sup>15</sup>. Quan es carrega la vista les dades s'incorporen automàticament amb les dades que s'han anat passant de vista en vista durant el procés de selecció de les dades.

---

<sup>15</sup> Les tires de text són objectes anomenats UILabel



**Figura 4-9.** Vista Desar.

En un primer moment es volia poder desar el castell a la base de dades i donar la possibilitat de desar i generar dos Tweets a la vegada. El primer més general i el segon més personal. L'API de Twitter quasi obliga a gestionar la creació de tweets des del primer pla de l'aplicació Tot i que és possible generar i publicar tweets des del fons de l'aplicació utilitzant la biblioteca Accounts.h, Apple i Twitter ho desaconsellen.

D'aquesta manera és impossible generar dos tweets a la vegada i que es visualitzin en primer terme abans que l'usuari doni el vist-i-plau i els envii. A més generar els Tweets i enviant-los amb el consentiment de l'usuari de l'aplicació també permet a aquest personalitzar el Tweet a enviar.

És per aquesta raó que finalment es va decidir crear tres possibles interaccions en aquesta pantalla. Únicament desar el castell, desar i generar un Tweet general i únicament generar un Tweet personal.

Si l'usuari de l'aplicació decideix només desar el castell, la classe encarregada de gestionar la vista cerca primer de tot el Twitter i la pàgina web de la colla per afegir-ho a la base de dades. Després es posa en contacte amb la base de dades, crea un objecte de tipus Castell i el desa. És una operació senzilla i que no comporta grans complicacions.

Un cop desat el castell l'aplicació obre una pantalla amb tots els castells desats fins el moment que permeten a l'usuari de l'aplicació comprovar que tot ha funcionat correctament. Des d'aquesta nova vista es possible esborrar qualsevol castell desat a la base de dades. Per fer-ho és necessari fer un swipe to delete i que consisteix en moure el dit per sobre d'una fila de dreta a esquerra. Un cop fet apareix un botó vermell anomenat



Delete que permet eliminar una entrada. L'eliminació no té marxa enrere i és automàtica. Un cop feta la taula amb la llista de castells fets i desats s'actualitza automàticament.



**Figura 4-10.** Exemple de swipe to delete.

Si l'usuari de l'aplicació decideix desar i generar un Tweet més general és necessari que cliqui a la fila anomenada Desar + Tweet colla de la vista Desar. Un cop fa clic l'aplicació executa totes les accions que es farien si es fes clic únicament a Desar i a més genera el tweet general.



**Figura 4-11.** Detall Tweet colla.

Com que aquest tweet és curt i impossible que sobrepassi els 140 caràcters màxims de Twitter només s'ha implementat un constructor del Tweet.

Si l'usuari de l'aplicació decideix generar un Tweet personal el castell seleccionat no es desa a la base de dades i només es genera el Tweet. És important destacar que la generació del tweet personal comporta la modificació de múltiples aspectes de les dades seleccionades que permeten evitar errors de gènere i número.

Com que les parts de text que permeten construir el Tweet són de mides variables<sup>16</sup> i els Tweets no poden tenir més de 140 caràcters ha sigut necessari crear diferents constructors del Tweet. D'aquesta manera si el Tweet generat amb el primer constructor té més de 140 caràcters es construeix un nou tweet amb el segon constructor, així fins que el Tweet té menys de 140 caràcters.



**Figura 4-12.** Quatre exemples de Tweets personals amb diferent text.

També s'han introduït algunes alertes importants per evitar errors, així doncs si l'usuari no va fer cas i intenta desar un castell sense haver seleccionat cap colla, apareix una alerta avisant del fet i el castell no es desa a la base de dades.



**Figura 4-13.** Alerta: selecciona colla i castell no desat.

Si l'usuari intenta generar un Tweet, personal o general, i no té configurat cap compte de Twitter al seu dispositiu mòbil apareix una alerta i el Tweet no es genera.

<sup>16</sup> Per exemple, el poble més curt te 3 caràcters i el més llarg 44.



**Figura 4-14.** Alerta: no tens configurat el compte a Twitter.

Si l'usuari intenta generar un Tweet personal i no ha seleccionat cap colla apareix una alerta i el Tweet no es genera.



**Figura 4-15.** Alerta: selecciona colla i Tweet no generat.

Per fer anar d'una manera correcta i fàcil el sistema de persistència, l'aplicació utilitza la classe CoreDataTableViewController. Aquesta classe ha estat creada per Paul Hegarty pel curs Developing Apps for iOS - CS193P de la Universitat d'Stanford<sup>17</sup>. Aquesta classe és una subclasse de UITableViewController i afegeix les funcionalitats necessàries per poder gestionar la persistència de dades.

### 4.3. Subsistema d'informació

L'estructura de la primera vista d'aquest subsistema presenta quatre botons que permeten accedir a les diferents vistes que implementen aquest subsistema. Les pantalles Què són els castells? I Vocabulari casteller estan implementades, al igual que altres vistes que hem vist anteriorment, sobre taules personalitzades. Per fer-ho ha estat necessari incloure requadres de text a les files i modificar l'aparença. D'aquesta manera s'han modificat els colors, el contorn dels requadres i l'ombra.

<sup>17</sup> <http://itunes.apple.com/itunes-u/ipad-iphone-application-development/id473757255>

Les definicions del vocabulari casteller han estat extretes de la Viquipèdia<sup>18</sup> i de la pàgina web dels Castellers de Sants<sup>19</sup>.



**Figura 4-16.** Vistes Què són els castells? i Vocabulari casteller.

Tal i com es va decidir en les primeres fases de desenvolupament de l'aplicació la vista amb informació detallada sobre les colles només implementa la informació sobre setze colles de les més de seixanta que existeixen actualment. Per seleccionar quines colles s'inclouen en l'aplicació en aquesta primera versió s'ha optat per consultar el rànquing de colles de l'any 2011 i agafar les primeres. Per fer-ho, i donat que existeixen múltiples rànquings, s'ha decidit utilitzar el de la pàgina web WebCasteller<sup>20</sup>.

Així doncs la vista anomenada Les colles està composta de 16 botons que contenen cada un d'ells l'escut d'una colla. Depenent del botó i l'escut que es selecciona es carrega una informació diferent a la pantalla de detall. Aquesta vista de detall conté informació bàsica de cada colla: localitat, adreça, color de la camisa, any de fundació, millor castell, millor actuació, diades castelleres importants, pàgina web i Twitter de la colla.

Tant la pàgina web com el Twitter de la colla són interactius i permeten obrir la pàgina web en el navegador Safari del dispositiu mòbil i enviar un Tweet a la colla utilitzant el diàleg de l'API de Twitter.

<sup>18</sup> [http://ca.wikipedia.org/wiki/Glossari\\_de\\_termes\\_castellers](http://ca.wikipedia.org/wiki/Glossari_de_termes_castellers)

<sup>19</sup> <http://www.borinots.cat/>

<sup>20</sup> <http://www.webcasteller.com/ca/>



**Figura 4-17.** Vista Les colles, el detall, generació de Tweet i pàgina web.

Com en altres llocs de l'aplicació, si l'usuari de l'aplicació intenta enviar un Tweet i no té configurat cap compte de Twitter al seu dispositiu mòbil apareix una alerta.

Finalment la vista que mostra On són les colles? és de les vistes que ha donat més feina a l'hora d'implementar-la, ja que ha estat necessari cercar les adreces de totes les colles una per una, cercar les localitzacions a GoogleMaps, agafar les coordenades i generar els punters que apareixen al mapa<sup>21</sup>.



**Figura 4-17.** Dos exemples de la vista On són les colles?

<sup>21</sup> Els mapes són objectes anomenats MKMapView

## 4.4. Subsistema d'estadístiques

Aquest subsistema incorpora, de manera general tres opcions: seleccionar paràmetres per filtrar les estadístiques, generar les estadístiques i exportar les dades.

Per fer la selecció de paràmetres s'ha creat una estructura de vistes delegades que fa que quan se selecciona un any o un resultat torni automàticament a carregar-se la vista d'estadístiques.

La selecció de paràmetres i anys permet seleccionar qualsevol dels que hi ha a la llista. A més a més, si l'usuari selecciona Tots, no es filtra res i es carreguen tots els castells sense tenir en compte l'any i/o el resultat.



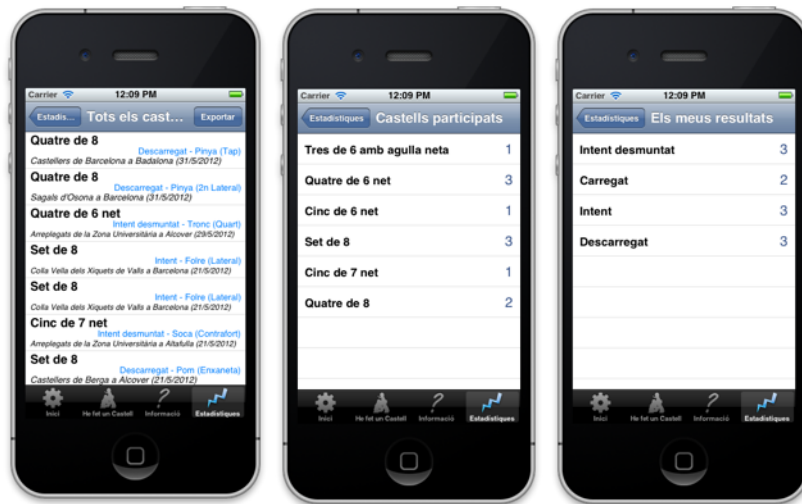
**Figura 4-18.** Vista principal, les dos vistes de filtratge i els filtres seleccionats.

Pel que fa a la generació d'estadístiques primerament es permet a l'usuari visualitzar tots els castells realitzats i desats a la base de dades. La pantalla que visualitza tots els castells és la mateixa que es visualitza quan es desa a la base de dades el castell, però afegeix la funcionalitat d'exportar les dades.

També és possible generar estadístiques que permeten visualitzar els castells participats i els resultats aconseguits. Per visualitzar aquestes estadístiques el programa també crida a la base de dades i recupera tots els castells que hi ha desats. Posteriorment els recorre i els conta per cada un dels paràmetres demanats.

S'ha decidit ser fidel a les especificacions inicials i només implementar les dos estadístiques que es proposaven al principi. Donat que generar més estadístiques és una feina pràcticament trivial

abans de posar l'aplicació a l'App Store d'Apple es generaran noves estadístiques que seran presentades amb el mateix format. Per exemple es poden generar estadístiques del tipus: les meves posicions i ciutats on he fet castells.



**Figura 4-19.** Les tres estadístiques que es poden generar.

Finalment, i pel que respecte a l'exportació de dades que hi ha emmagatzemades al dispositiu mòbil, un cop es clica el botó que hi ha a la barra de navegació anomenat Exportar, la classe fa una crida a la funció que s'encarrega de generar una presentació en html i presentar-la en un correu electrònic que l'usuari de l'aplicació pot enviar qualsevol adreça desitjada.



**Figura 4-20.** Exportar.

## 5. Conclusions

La conclusió més important a la que s'ha d'arribar és que s'ha aconseguit pensar, dissenyar i implementar una aplicació per iPhone de principi a fi. S'ha de tenir molt en compte que s'han assolit tots els objectius inicialment proposats i descrits en els primers documents. També és important tenir present que el producte proposat representava un grau de complexitat prou elevat per un desenvolupador que no tenia cap experiència personal en el desenvolupament d'aplicacions mòbils. A aquesta falta d'experiència s'havia de sumar el baix coneixement del llenguatge de programació Objective-C, de l'entorn Apple, de l'IDE de desenvolupament Xcode i del Framework principal per desenvolupar en iOS Cocoa Touch.

Per fer el projecte .Castells ha estat molt important l'aprenentatge incremental. Com exemple es pot dir que en un primer moment es van dibuixar en algunes pantalles simples botons que permetien generar accions. Un cop es van aprendre altres possibilitats per generar accions i es va trobar el lloc adequat per inserir-les es van haver de modificar algunes parts de l'aplicació.

Un bon exemple pot ser la primera pantalla del Subsistema d'informació. En un primer moment estava dissenyada com una vista amb 4 botons<sup>22</sup>. Finalment es va decidir implementar una taula amb 4 cel·les estàtiques<sup>23</sup> que fan la funció de botons però que permeten un millor disseny.



**Figura 5-1.** Comparació vista Informació entre el prototip i la definitiva.

<sup>22</sup> Els botons són objectes anomenats UIButton.

<sup>23</sup> L'IDE Xcode de desenvolupament les anomena Static Cells.



Aquestes decisions van comportar les primeres operacions de redisseny de l'aplicació. Tot i que un cop après a fer-ho no va ser difícil aplicar els canvis, si que va ser interessant adonar-se que el desenvolupament ha estat totalment incremental.

Han sigut molt importants durant aquest últims mesos els tutorials per desenvolupar en iOS que es poden trobar a Internet. S'ha d'estar molt agraït a la gent que inverteix el seu temps en preparar eines que permeten a altres desenvolupadors aprendre a programar aplicacions per iPhone i iPad.

La part més feixuga ha estat trobar els tutorials i els exemples que es podien utilitzar. En aquest sentit no només era important trobar un tutorial ben explicat, també ha estat important trobar que el que explicava es pogués traslladar d'alguna manera a l'aplicació que s'estava desenvolupant.

Finalment ha estat necessari lluitar contra el constant canvi de la tecnologia. Ha estat un problema trobar tutorials i guies actualitzades a les últimes actualitzacions del sistema operatiu iOS 5.1 i a la última versió de l'IDE per desenvolupar aplicacions per iPhone Xcode 4.3.1.

Per ajudar a qualsevol nou desenvolupador que vulgui començar a programar aplicacions per iPhone i a la vegada per ajudar-me a mi mateix per tenir en algun lloc una espècie de reservori de bons tutorials i mètodes vaig crear un blog<sup>24</sup> on s'han anat creant entrades un cop es descobrien tutorials o bona informació que era utilitzada per desenvolupar l'aplicació.



Figura 5-2. Pantalla principal del blog creat.

<sup>24</sup> <http://aprendreios.wordpress.com/>

És molt important percebre com les primeres entrades són de tutorials molt bàsics i les últimes, moltes vegades, són de simples mètodes o crides que permeten, per exemple, obrir el navegador web safari des de dins de l'aplicació.

D'altra banda, ha estat molt important l'autodisciplina i el treball constant per tal de poder complir amb els requisits temporals que exigia el projecte.

Des del punt de vista personal el més important era aconseguir una mínima experiència professionalitzadora que hem permetès encaminar el futur professional cap al desenvolupament d'aplicacions mòbils per iPhone, iPad i iPod. En aquest sentit és evident que fins els propers mesos no es podrà tenir la certesa d'haver aconseguit l'objectiu, però tot indica que les perspectives laborals que s'obren davant meu són prou bones.

En uns moments en els que Objective-C està a punt de situar-se en el tercer lloc dels llenguatges més utilitzats per davant de C++ i només per darrera dels imbatibles C i Java és molt important haver aconseguit uns coneixements mínims i una pràctica molt interessant<sup>25</sup>.

De la mateixa manera en el moment d'explosió respecte a la demanda d'aplicacions mòbils en la que ens trobem, sembla molt apropiat haver triat aquesta àrea de Treball Final de Carrera per tal de poder encaminar el meu futur professional. El volum de demanda de programadors iOS està en constant creixement i el volum de professionals que es dediquen a la programació en aquest entorn és encara avui dia prou baix.

## 5.1. Línies de futur

És important destacar dues línies de futur diferents. Per un costat aquelles modificacions que seria interessant fer a l'aplicació abans d'enviar-la a l'App Store d'Apple. Algunes s'han comentat amb anterioritat en aquest document i altres no.

És important dir que aquests canvis no comporten moltes hores de feina però que no s'han aplicat abans perquè la majoria s'han detectat una vegada l'entregable ja estava en mans del consultor. S'ha d'agrair molt especialment a quatre persones que han testejat l'aplicació utilitzant el programa TestFlight i que han aportat alguna de les idees:

- Eliminar la imatge inicial quan es carrega l'aplicació, de fet Apple no recomana utilitzar-la i pel fet que dura menys d'un segon a la pantalla del dispositiu crea frustració a l'usuari que no pot tornar a veure-la.

---

<sup>25</sup> <http://www.tiobe.com/index.php/content/paperinfo/tpci/index.html>

- Afegir les colles que no pertanyen a la Coordinadora de Colles Castelleres de Catalunya. S'han d'afegir tant al llistat de colles com al mapa que les situa arreu del mon.
- Afegir més estadístiques. Per exemple es poden generar estadístiques del tipus: les meves posicions i ciutats on he fet castells.
- Eliminar el primer Tab Bar. Aquesta eliminació inclou eliminar la vista d'Instruccions i recol·locar la vista Sobre .Castells.
- No deixar fer res si l'usuari no ha seleccionat colla. S'ha de deixar encara més clar que si no es selecciona colla no es pot continuar amb la selecció de paràmetres. La solució ha de ser desconnectar el botó que ens porta a la següent pantalla si no s'ha seleccionat colla.
- Arreglar alguns detalls de presentació respecte els fons de les diferents vistes.

D'altra banda, respecte a les línies de futur a més llarg plaç es poden destacar tres de molt importants, però que ara mateix es troben per sobre de les capacitats del programador de l'aplicació i que per temps era impossible plantejar dins d'aquest TFC.

- Crear una pàgina web i una base de dades externa que permeti emmagatzemar tots els castells fets pels usuaris de l'aplicació. Aquesta pàgina web a la vegada, hauria de permetre visualitzar les mateixes estadístiques personals que es poden veure a l'iPhone més unes noves estadístiques generals de tots els castells desats per tots els usuaris.
- Internacionalitzar l'aplicació traduint-la a diferents idiomes. Principalment al castellà, al portuguès brasiler, al xinès i l'anglès. La raó de triar aquests idiomes ve donada perquè existeixen colles castelleres a Xile, Brasil, EUA i la Xina.
- Afegir la totalitat de colles conegudes a la vista Les colles.

## 6. Glossari

---

**Apple** Empresa nord-americana de tecnologia líder en el sector que comercialitza, entre d'altres, els dispositius iPad i iPhone i els ordinadors Mac.

**AppStore** Botiga oficial on es comercialitzen totes les aplicacions que es desenvolupen per dispositius mòbils de la companyia Apple. Tot i que hi ha altres maneres, al llindar de la legalitat, d'obtenir aplicacions, l'única manera oficial de fer-ho és en aquesta botiga.

**Cocoa Touch** Interfície de programació d'aplicacions propietat d'Apple que serveix per facilitar el desenvolupament d'aplicacions per iPad i iPhone. Està formada per un conjunt de classes, funcions, procediments i mètodes principalment fent servir una tecnologia orientada a objectes.

**Core Data** Llibreria inclosa en l'entorn de desenvolupament d'aplicacions d'Apple que serveix per gestionar les relacions entre objectes i la seva representació en fitxers, permetent al desenvolupador no haver d'entrar en detalls relacionats amb bases de dades.

**Framework** Marc de treball que s'utilitza en el desenvolupament de programari i que inclou tot un conjunt d'eines, llibreries i recursos pel programador que li faciliten la seva tasca.

**Internacionalització** Procés que permet que un programari estigui disponible en diversos idiomes, pensant especialment en els textos i avisos que s'hi mostren. Sovint es representa amb l'abreviatura i18n.

**IDE** Programari utilitzat en la creació d'aplicacions consistent en un entorn de desenvolupament integrat que acostuma a incloure, com a mínim, facilitats per escriure codi font, compilar-lo i depurar-ne els errors.

**iOS** Sistema operatiu que utilitzen els dispositius mòbils desenvolupats per l'empresa Apple, com per exemple l'iPad.

**MVC** Patró de disseny que consisteix en separar de manera clara en una aplicació les classes de negoci i dades (Model), la interfície gràfica que es mostra a l'usuari (Vista) i la lògica de control que enllaça una part amb l'altra (Controlador).

**Objective-C** Llenguatge de programació orientat a objectes que s'utilitza pel desenvolupament d'aplicacions pels diferents sistemes operatius de l'empresa Apple. Es tracta d'un llenguatge creat sobre la base del llenguatge C, per la qual cosa permet fer ús de qualsevol codi font preexistent en aquest.

**persistència** Característica que permet que les dades que formen part d'un programari no desapareguin quan aquest deixa d'executar-se, sinó que s'emmagatzemin de tal manera que puguin ser recuperades la propera vegada que es fa servir el programari.

**SQLite** Sistema gestor de bases de dades que destaca pels seus requeriments mínims de recursos i per la senzilla instal·lació i funcionament en qualsevol maquinari, aspecte que el fa òptim per la seva utilització com a base de dades en dispositius mòbils.

**Xcode** IDE específic de la plataforma Mac que facilita el desenvolupament d'aplicacions per dispositius de l'empresa Apple.

## 7. Bibliografia

---

**Andreas** (2009). UIButton in UITableView Footer [en línia]. <http://blog.blackwhale.at/?p=104> [data consulta: 20/05/2012].

**Apple** (2011). Core Data Programming Guide. Cupertino, CA, USA: Apple Inc.

**Apple** (2010). Core Data Tutorial for iOS. Cupertino, CA, USA: Apple Inc.

**Arnaut, Dani** (2011). iOS 5 development. [en línia]. [http://www.youtube.com/watch?v=\\_YZYKg8hWD8](http://www.youtube.com/watch?v=_YZYKg8hWD8)

**Carmine** (2012) [iOS SDK] Add UISearchBar to UITableView programmatically. [en línia]. <http://www.minux.eu/blog/index.php/2012/02/01/ios-sdk-add-uisearchBar-to-uitableview-programmatically/> [data consulta: 20/05/2012].

**Conway, Joe; Hillegass, Aaron** (2010). Desarrollo de Aplicaciones para iPhone & iPad. Madrid: Anaya Multimedia.

**Diversos autors** (2012). Viquipèdia: l'enciclopèdia de contingut lliure. [en línia]. <http://ca.wikipedia.org/> [data consulta: 20/05/2012].

**Diversos autors** (2012). Wikipedia: the free encyclopedia. [en línia]. <http://en.wikipedia.org/> [data consulta: 20/05/2012].

**Diversos autors** (2012). StackOverflow. [en línia]. <http://stackoverflow.com/> [data consulta: 20/05/2012].

**Diversos autors** (2012). iPhone Dev SDK. [en línia]. <http://www.iphonedevsdk.com/> [data consulta: 20/05/2012].

**Duff, John** (2010). Building a SearchView with UISearchBar and UITableView. [en línia]. <http://jduff.github.com/2010/03/01/building-a-searchview-with-uisearchBar-and-uitableview/> [data consulta: 20/05/2012].

**Hegarty, Paul; Stanford University** (2011). Developing Apps for iOS CS193P [en línia]. <http://itunes.apple.com/itunes-u/ipad-iphone-application-development/id473757255> [data consulta: 20/05/2012].

**Holleman, Matthijs** (2012). Beginning Storyboards in iOS 5. [en línia]. <http://www.raywenderlich.com/5138/beginning-storyboards-in-ios-5-part-1> [data consulta: 20/05/2012].

**Izquierdo, Jorge** (2011). Aprende a programar aplicaciones para iOS. [en línia]. <http://www.youtube.com/watch?v=PngmajbYRU8>

**López Hernández, Fernando** (2012). Objective-C. Curso práctico para programadores Mac OS X, iPhone y iPad. Madrid: RD Libros.

**Nahavandipoor, Vanda** (2012). iOS 5 Programming Cookbook. Sebastopol, CA, USA: O'Reilly Media.

**Privat, Michael; Warner, Robert** (2011). Pro Core data for iOS. New York: Apress.

**Ray, John; Johnson, Sean** (2010). Desarrollo de aplicaciones para iPhone. Madrid: Anaya Multimedia.

**Roadley, Tim** (2012). Core Data Basics. [en línea]. <http://timroadley.com/2012/02/09/core-data-basics-part-1-storyboards-delegation/> [data consulta: 20/05/2012].

**Rodríguez, M. Elena; Sistac, Jaume** (2011). Base de dades I: Mòdul 6: Teoria de la normalització. Barcelona: Fundació UOC.

**Techotopia** (2012). An iOS 5 iPad Core Data Tutorial. [en línea]. [http://www.techotopia.com/index.php/An\\_iOS\\_5\\_iPad\\_Core\\_Data\\_Tutorial](http://www.techotopia.com/index.php/An_iOS_5_iPad_Core_Data_Tutorial) [data consulta: 20/05/2012].

**Wenderlich, Ray; Burkepile, Adam** (2012). Core Data on iOS 5 Tutorial: Getting Started. [en línea]. <http://www.raywenderlich.com/934/core-data-on-ios-5-tutorial-getting-started> [data consulta: 20/05/2012].

**ZButton** (2011). Desmitificando Core Data para iOS. [en línea]. <http://zbutton.wordpress.com/2011/03/06/desmitificando-core-data-para-ios/>